Segmentation multi-vues par coupure de graphes

Abdelaziz Djelouah, Jean-Sébastien Franco, Edmond Boyer, François Le Clerc, Patrick Pérez

Segmentation multi-vues par coupure de graphes

A.Djelouah[1,2] J-S.Franco[2] E.Boyer[2] F.Le Clerc[1] P.Pérez[1]

[1] Technicolor R&I [2] LJK - INRIA Rhône-Alpes

Résumé

Dans cet article, nous abordons le problème de la segmentation simultanée d'images lorsque plusieurs caméras calibrées et synchronisées observent la même scène. Nous proposons une nouvelle approche permettant de propager l'information de segmentation de manière cohérente entre les vues. Pour cela, le problème de segmentation est formulé comme un problème d'étiquetage en deux régions fond et forme des pixels de l'image, résolu avec une méthode de coupe de graphe. Contrairement à de nombreuses approches de l'état de l'art, notre méthode ne nécessite pas de reconstruction 3D dense de l'objet mais plus simplement un échantillonnage éparse de l'espace 3D. Une évaluation complète est effectuée sur des données statiques standard. Les résultats obtenus montrent l'intérêt de la méthode qui obtient des résultats équivalents à ceux de l'état de l'art mais avec beaucoup moins de points de vue.

Mots Clef

Segmentation, segmentation multi-vues.

Abstract

In this paper, we address the problem of object segmentation in multiple views when two or more viewpoints of the same scene are available. We propose a new approach that propagates segmentation coherence information in space, hence allowing evidence in one image to be shared over the complete set. To this aim the segmentation is cast as a single efficient labeling problem over space and time with graph cuts. In contrast to most existing multi-view segmentation methods that rely on some form of dense reconstruction, ours only requires a sparse 3D sampling to propagate information between viewpoints. The approach is thoroughly evaluated on standard multi-view datasets. The obtained results compete with state of the art methods but they are achieved with significantly fewer viewpoints.

Keywords

Segmentation, multi-view segmentation.

1 Introduction

La segmentation des objets d'intérêt dans les images est une étape préliminaire à de nombreuses applications de vision par ordinateur : matting, compositing, correction de couleur, indexation d'images, et, plus généralement, l'analyse d'images. La possibilité de segmenter de manière automatique un même objet lorsque plusieurs images de cet objet sont disponibles suscite de plus en plus d'intérêt car elle élimine le besoin de connaissances a priori sur la forme ou sur l'apparence de l'objet, ainsi que le besoin d'interaction avec l'utilisateur [17] comme c'est le cas actuellement avec les séquences monoculaires. Ce travail a pour objectif la segmentation automatique d'un objet à partir d'images d'au moins deux caméras calibrées. Comme remarqué dans [22], ce type de problème est intrinsèquement difficile, en particulier lorsqu'il y a peu d'images et que les points de vue sont éloignés. Dans ce cas, l'hypothèse traditionnelle d'une apparence similaire de l'objet dans plusieurs vues devient irréaliste. A cet égard, le problème de la segmentation multi-vues est différent de celui de la cosegmentation [18, 11], qui fait l'hypothèse que les avant-plans partagent la même apparence entre les images alors que les arrière-plans sont différents d'une image à l'autre. Notre approche suppose simplement que l'objet d'intérêt est présent dans toutes les images considérées.

FIGURE 1 – Résultats de notre approche de segmentation multi-vues sur 3 images, sans aucune interaction utilisateur ou hypothèse de photo-cohérence.

Notre approche suppose aussi que les caméras sont étalonnées pour imposer des contraintes de cohérence de formes entre les points de vue. Cet étalonnage est souvent disponible dans les applications où plusieurs caméras sont utilisées, ou il peut être facilement estimé avec des outils tels que Bundler [20]. C'est le cas notamment dans les studios d'enregistrement [8] ou pour la vidéo-surveillance mais aussi avec des scénarios plus complexes qui font intervenir des caméras mobiles [9].

Nous proposons une nouvelle formulation itérative (§4) pour effectuer la segmentation multi-vues (Fig. 1) qui utilise un graphe liant les pixels entre vues. Cette formulation s'inspire en partie des outils développés par la communauté de la cosegmentation pour corréler la segmentation dans plusieurs vues [10, 21]. Elle diffère cependant par le fait d'utiliser des contraintes géométriques plutôt que photométriques. Les principales contributions de notre approche sont les suivantes : (a) la représentation sous forme de graphe des contraintes du problème de segmentation multi-vues. (b) Une convergence rapide et une complexité moindre grâce à des contraintes multi-vues éparses. (c) La capacité de traiter des situations avec peu de points de vue, très espacés les uns des autres, une situation qui apparaît naturellement en pratique mais qui est très peu considérée par les méthodes de l'état de l'art.

2 État de l'art

2.1 Segmentation Multi-vues

Le problème apparait pour la première fois dans les travaux de Zeng *et al.*[24] où une solution rudimentaire est proposée, basée sur l'utilisation des silhouettes de l'objet pour construire un objet 3D cohérent. Plusieurs méthodes suivront par la suite cette voie en faisant une reconstruction 3D explicite de l'objet en alternance avec une segmentation dans l'image basée sur les modèles d'apparence fond/forme [5, 15, 8, 16].

D'autres représentations de l'objet d'intérêt sont aussi utilisées, le plus souvent basées sur les silhouettes ou le volume d'occupation [5], sur la profondeur [7, 8], ou la stéréo [13]. Tout un ensemble de techniques est utilisé pour régulariser l'occupation spatiale comme des coupes de graphe [5, 8], ou de l'optimisation globale [12]. Une proportion significative de l'état de l'art nécessite de l'interaction utilisateur [12, 23].

De nombreuses approches reconstruisent un modèle 3D de l'objet, la segmentation étant alors obtenue par reprojection. Il y a néanmoins une motivation indéniable à éviter la reconstruction 3D dense lorsqu'il s'agit de travailler sur un faible nombre de points de vue : les modèles 3D obtenus à partir d'images n'atteignent en effet une qualité satisfaisante qu'à partir d'une douzaine de vues. Ainsi, une approche récente [13] utilise un très grand nombre de caméras avec un faible espacement. Notre objectif est d'atteindre une qualité équivalente avec un minimum de points de vue, aussi éloignés que possible les uns des autres. Dans cet objectif nous nous focalisons sur la manière de propager l'information entre les vues pour obtenir une étiquetage cohérent des pixels, et non pas sur une modélisation 3D précise de l'objet.

Imposer la cohérence géométrique d'une vue à l'autre s'est révélé être une opération difficile. En effet, les contraintes qui s'expriment simplement en 3D, se traduisent en contraintes beaucoup plus complexes avec les droites épipolaires, *e.g.* [14, 19]. Dans [4] un schéma de coupe de graphe de superpixels est utilisé avec des contraintes justement dérivées de la géométrie épipolaire conjointement avec de la stéréo. Ce type de méthodes requiert des caméras positionnées en demi-cercle avec un faible espacement, comme dans [13], et le recours à des approches heuristiques afin de limiter le nombre de liens créés entre plusieurs vues.

Nous nous inspirons de [6] qui utilise un échantillonnage épars du volume 3D. Cette approche se révèle être une alternative efficace à la reconstruction dense. Dans cet article nous proposons une construction de graphe qui permet le transfert d'information entres vues de manière efficace.

2.2 Approches de Cosegmentation

Dans [18], la cosegmentation est définie comme la segmentation binaire simultanée de régions dans une paire d'images, et par extension dans plusieurs images [2, 11, 22]. L'hypothèse clé de ces méthodes est l'observation d'une région "avant-plan" commune, ou d'objets partageant une apparence commune contrairement à un arrière plan qui lui varie fortement d'une image à l'autre. Comme noté par [22], la cosegmentation fait référence à des scénarios de plus en plus divers allant de la segmentation guidée par utilisateur à la segmentation de classes d'objets plutôt que d'instances. Le problème de segmentation multi-vues diffère par le fait que seules les contraintes géométriques sont prises en compte.

3 Principe

Suivant en cela [14], nous définissons l'objet d'avant plan comme l'objet entièrement vu par toutes les caméras et dont l'apparence générale est différente de celle de l'arrière plan. Le problème de segmentation est formulé comme un problème d'étiquetage conjoint parmi les n vues, régi par une seule énergie de type champ aléatoire de Markov discutée en §4. Premièrement, pour assurer la propagation d'information entre les vues, nous nous appuyons sur l'idée qu'un échantillonnage 3D épars dans la région d'intérêt (le champ de vision commun de toutes les caméras) produit suffisamment de liens entres les images [6]. Chaque échantillon crée un lien entre lui et les pixels où il se projette, dont la force dépend de la probabilité de l'échantillon d'appartenir à l'objet. Deuxièmement, pour assurer une propagation intra-vue efficace, l'image est sur-segmentée en superpixels et deux ensembles de voisinages sont définis : l'un dans l'espace image et le second dans l'espace texture. Utiliser des superpixels permet de bénéficier d'une caractérisation des régions plus riche, réduisant ainsi les ambiguïtés d'apparence. Troisièmement, l'énergie résultante est minimisée en faisant une coupe de graphe [3] et le résultat est utilisé pour ré estimer les modèles d'apparence et les probabilités fond/forme associées aux échantillons 3D. Les détails de chaque étape sont présentés ci-après.

4 Formulation

Soit $I = \{I^1, ..., I^n\}$ l'ensemble des n images provenant des n vues. Pour chaque image i on note $\mathcal{P}_i$ l'ensemble

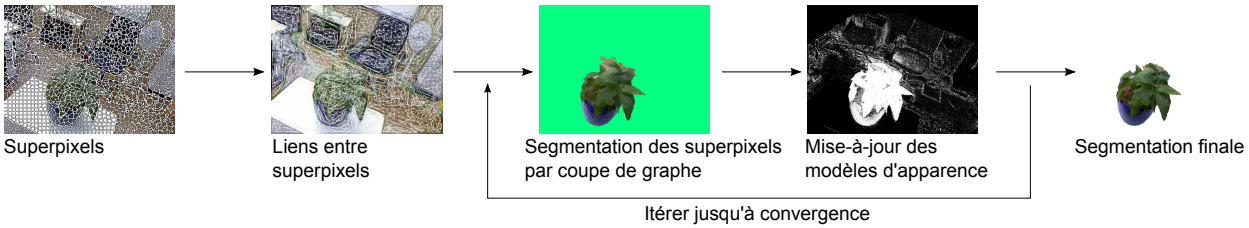


FIGURE 2 – Aperçu global : Les superpixels sont estimés en utilisant la méthode SLIC [1]. Les liens entre superpixels (en blanc) sont calculés en utilisant les descripteurs de superpixels. Le processus itératif alterne ensuite entre coupe de graphe et mise-à-jour des modèles d'apparence. La segmentation au niveau pixel est estimée à la fin du processus itératif.

de ses superpixels p. Segmenter l'objet d'intérêt revient à trouver pour chaque superpixel $p \in \mathcal{P}_i$ son étiquette x_p avant plan ou arrière plan : $x_p \in \{\mathrm{f}, \mathrm{b}\}$. $\mathcal{S}$ est l'ensemble des points 3D qui sont échantillonnés de manière uniforme dans le volume de visibilité commun.

4.1 Énergie du champ aléatoire de Markov

Étant donné la décomposition en superpixels et les échantillons 3D (voir Fig. 3), nous souhaitons que l'énergie proposée favorise un étiquetage qui suive les principes suivants :

Apparence individuelle L'apparence de chaque superpixel doit respecter le modèle avant-plan ou arrière-plan de sa vue, en fonction de son étiquette.
Continuité d'apparence. Les superpixels voisins dans l'image présentant une même apparence sont plus susceptibles d'avoir la même étiquette.
Similarité d'apparence. Deux superpixels avec une même apparence (couleur/texture) font probablement partie d'un même objet mais ne sont pas forcément voisins, en raison d'occultations éventuelles. Ces superpixels sont plus susceptibles d'être classés de la même manière.
Cohérence multi-vues. Les échantillons 3D sont considérés comme faisant partie de l'objet s'ils se projettent dans les régions avant-plan dans toutes les vues.
Contrainte de projection. En supposant qu'il existe un nombre suffisant d'échantillons 3D, un superpixel devrait être avant-plan si au moins un échantillon 3D appartenant à l'objet se projette sur lui.

4.2 Termes d'apparence intra-vue

Nous utilisons ici les termes unaires et binaires classiques. On notera toutefois que pour le terme binaire on considère deux types de voisinages : le premier dans l'espace image et le second dans l'espace des descripteurs d'apparence.

Terme d'apparence individuel. On note E_c le terme unaire d'attache aux données lié à l'apparence de chaque superpixel. L'apparence est caractérisée par les probabilités d'être prédit par les modèles fond ou forme comme suit :

$$E_c(x_p) = \begin{cases} \sum_{r \in \mathcal{R}_p} -\log H_i^B(I_r^i) & \text{si } x_p = \mathrm{b}, \\ \sum_{r \in \mathcal{R}_p} -\log H_i^F(I_r^i) & \text{si } x_p = \mathrm{f}, \end{cases} \quad (1)$$

où $\mathcal{R}_p$ est l'ensemble des pixels appartenant au superpixel p. Une combinaison d'histogrammes de couleurs et de textures est utilisée pour modéliser l'apparence. Dans notre cas, I_r^i est un vecteur à 11 dimensions qui inclut couleur et texture. La texture est définie comme étant l'amplitude du gradient sur 4 échelles d'image et celle du Laplacien sur 2 échelles. Comme étape d'initialisation, un algorithme des k-moyennes est exécuté séparément sur les valeurs couleur et les valeurs de texture. Cette étape de regroupement permet de créer un vocabulaire de descripteurs sur lequel sont construits les histogrammes (H_i^F et H_i^B).

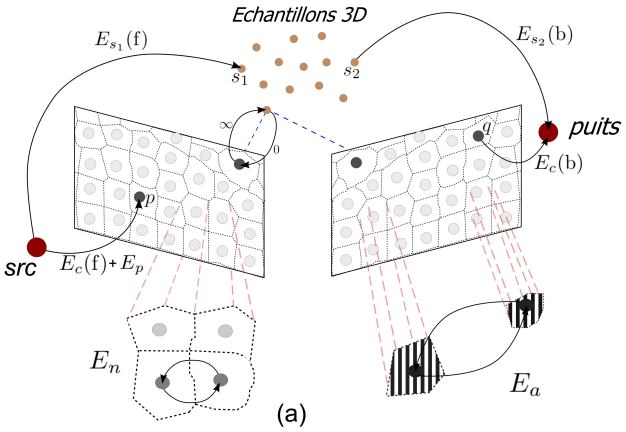


FIGURE 3 – Construction du graphe : les superpixels et les échantillons 3D sont les sommets de notre graphe. Les arrêtes contiennent les différents termes de l'énergie. Une coupe minimale dans ce graphe fournit une solution à notre problème de minimisation d'énergie.

Terme de continuité d'apparence. Ce terme binaire, noté E_n, pénalise l'affectation de différentes étiquettes à des superpixels voisins et d'apparence similaire. Pour modéliser cette similarité, on utilise des histogrammes sur le vocabulaire de descripteurs précédemment définis. L'histogramme constituant le descripteur d'apparence d'un superpixel p, est noté A_p. Soit $\mathcal{N}_n^i$ l'ensemble des paires de superpixels voisins dans la vue i. Pour chaque $(p, q) \in \mathcal{N}_n^i$, le terme E_n est inversement proportionnel à la distance entre les descripteurs des deux superpixels :

$$E_n(x_p, x_q) = \begin{cases} \exp\left(\frac{-d(A_p, A_q)^2}{2 < d(A_p, A_q)^2 >}\right) & \text{si } x_p \neq x_q, \\ 0 & \text{sinon.} \end{cases} \quad (2)$$

La distance $d(.,.)$ correspond ici à une distance χ^2 entre les descripteurs. $< d(A_p, A_q) >$ indique l'espérance sur

tous les superpixels voisins.

Terme de similarité d'apparence. Pour favoriser un étiquetage cohérent et une propagation efficace entre superpixels similaires, on introduit un second terme binaire E_a de la même forme que E_n. Ce terme E_a, non local, relie chaque superpixel p avec ses k plus proches voisins dans l'espace des descripteurs. On note $\mathcal{N}_a^i$ l'ensemble des paires de superpixels similaires suivant ce critère, et on définit :

$$E_a(x_p, x_q) = \begin{cases} \exp(\frac{-d(A_p,A_q)^2}{2<d(A_p,A_q>^2}) & \text{si } x_p \neq x_q, \\ 0 & \text{sinon.} \end{cases} \quad (3)$$

4.3 Termes de cohérence géométrique inter-vues

Pour propager l'information entres les vues, on utilise une structure de graphe connectant chaque échantillon 3D avec les superpixels sur lesquels il se projette. Cela revient à une construction similaire à [10] où un lien fort est créé entre les pixels ayant une apparence commune et où ce lien est maintenu tout au long de la procédure. Une différence clé dans notre approche est que l'influence des échantillons 3D peut changer au fur et à mesure des itérations. De ce fait la cohérence des échantillons est réévaluée après chaque itération, ce qui se traduit par une mise-à-jour des poids sur les arêtes du graphe reliant les échantillons 3D à la source et au puits.

Terme de cohérence pour les échantillons Soit P_s^{f} la probabilité de cohérence d'un échantillon 3D $s \in \mathcal{S}$. Cette probabilité P_s^{f} est calculée en utilisant les régions dans lesquelles se projette l'échantillon 3D de la même manière que dans [6]. Un terme d'énergie unaire et une étiquette x_s sont associés à chaque échantillon 3D. Cela permet de décider, à la volée, si un échantillon s fait partie de l'objet en utilisant tous les termes de l'énergie.

$$E_s(x_s) = \begin{cases} -\log(1 - P_s^{\mathrm{f}}) & \text{si} \quad x_s = \mathrm{b}, \\ -\log P_s^{\mathrm{f}} & \text{si} \quad x_s = \mathrm{f}. \end{cases} \quad (4)$$

Terme de jonction échantillon-superpixel. Pour assurer la cohérence, chaque échantillon s est connecté à tous les superpixels p sur lesquels il se projette, définissant un voisinage $\mathcal{N}_s$ de l'échantillon sur lequel le terme binaire E_j suivant s'applique (voir figure 4) :

$$E_j(x_s, x_p) = \begin{cases} \infty & \text{si } x_s = \mathrm{f} \text{ et } x_p = \mathrm{b}, \\ 0 & \text{sinon.} \end{cases} \quad (5)$$

Le principal intérêt de ce terme est d'empêcher toute coupe de graphe qui assignerait simultanément un superpixel p au fond et un échantillon 3D se projetant dessus à l'avant-plan. On a donc la propriété suivante : étiqueter un superpixel p comme fond n'est possible que s'il est cohérent d'assigner à la classe d'arrière-plan tous les échantillons 3D se projetant sur lui.

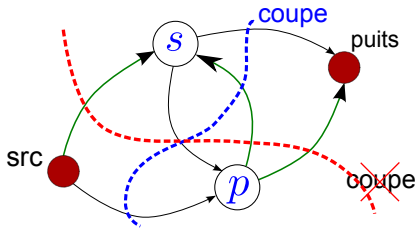


FIGURE 4 – Relation entre superpixel et échantillon. Si un échantillon est étiqueté comme avant-plan, alors les superpixels sur lesquels il se projette ne peuvent pas être étiquetés comme arrière-plan car cela correspond à une coupe avec un coût infini.

La propriété inverse qui est que : "pour être étiqueté avant-plan, un superpixel doit avoir au moins un échantillon 3D assigné à l'avant-plan sur sa ligne de vue" ne peut être obtenue qu'avec des termes d'énergie sur des champs aléatoires de Markov d'ordre supérieur. Pour avoir un comportement similaire, nous utilisons un terme unaire réévalué à chaque itération comme suit.

Terme de projection des échantillons. Le comportement désiré peut être approché en associant à chaque superpixel p un terme de projection $P(x_p|\mathcal{V}_p)$. Le but de ce terme est de pénaliser l'étiquetage de p comme avant plan si aucun échantillon n'a été étiqueté comme avant-plan dans la région 3D $\mathcal{V}_p$ vue par ce superpixel. Le terme d'énergie correspondant est défini par :

$$E_p(x_p) = -\log P(x_p|\mathcal{V}_p) \quad \text{où} \quad \mathcal{V}_p = \max_{s \in \mathcal{V}_p}(P_s^{\mathrm{f}}) \quad (6)$$

4.4 Énergie et construction du graphe

Soit X la conjonction de toutes les étiquettes pour les échantillons et les superpixels. L'énergie peut s'écrire sous la forme de 2 groupes de termes : groupe des termes intra-vue et groupe des termes inter-vues. $\lambda_1, \lambda_2, \lambda_3$ sont les poids associés aux différents termes d'énergie. Trouver une segmentation pour l'ensemble des vues, étant donné les histogrammes H_i^B et H_i^F et les probabilités P_s^{f}, revient à trouver l'étiquetage X qui minimise :

$$\begin{aligned} E(X) = \sum_i \Bigg[& \sum_{p \in \mathcal{P}_i} E_c(x_p) + \lambda_1 \sum_{(p,q) \in \mathcal{N}_n^i} E_n(x_p, x_q) \\ & + \lambda_2 \sum_{(p,q) \in \mathcal{N}_a^i} E_a(x_p, x_q) + \sum_i \sum_{p \in \mathcal{P}_i} E_p(x_p) \Bigg] \\ & + \sum_{s \in \mathcal{S}} \lambda_3 E_s(x_s) + \sum_{(s,p) \in \mathcal{N}_s} E_j(x_s, x_p) \end{aligned} \quad (7)$$

Étant donné que la contrainte de submodularité est satisfaite dans ce modèle, on peut construire un graphe G (de type source/puits) où la coupe minimale va donner la solution au problème de minimisation d'énergie. Ce graphe contient deux sommets terminaux *source* and *puits*,

un sommet pour chaque échantillon 3D et un autre pour chaque superpixel. Des arêtes sont ajoutées entre les sommets en fonction des termes d'énergie précédemment définis (voir Fig. 3 pour le graphe résultant).

5 Approche algorithmique

Comme la plupart des méthodes de segmentation de l'état de l'art, nous adoptons une stratégie itérative alternant entre la coupe de graphe précédemment décrite et une mise-à-jour des modèles d'apparence. La contrainte de visibilité commune de l'avant-plan peut être utilisée pour initialiser les modèles couleur, comme dans [14].
La Fig. 5 donne un aperçu global de l'approche. L'estimation des superpixels, des descripteurs et des liens se fait une seule fois lors de la phase d'initialisation. Dans le processus itératif, les termes unaires sont estimés en utilisant les modèles d'apparence obtenus à l'itération précédente. L'algorithme converge lorsqu'il n'y a plus de changement d'étiquette d'une itération à l'autre. La classification finale des superpixels est utilisée pour estimer la segmentation binaire avant-plan/arrière-plan au niveau pixel, au moyen d'une approche de coupe de graphe classique.

Initialisation

1. Segmenter l'image en superpixels.
2. Estimer les descripteurs pour tous les superpixels.
3. Lier les superpixels similaires.
4. Échantillonner uniformément l'espace 3D.
5. Initialiser les modèles d'apparence.

Étapes itérées

6. Estimer les termes unaires avec les nouveaux modèles.
7. Trouver la coupe de graphe qui minimise l'énergie.
8. Mettre à jour les modèles couleurs.

Finalisation

9. Segmentation finale : segmentation par coupe de graphe standard des pixels en utilisant les modèles d'apparence de la segmentation des superpixels.

FIGURE 5 – Aperçu global de l'algorithme.

6 Résultats expérimentaux

6.1 Protocole expérimental

L'approche a été implémentée en utilisant SLIC [1] comme algorithme de sur-segmentation de l'image en superpixels et l'implémentation de Kolmogorov de la coupe de graphe [3]. La taille des superpixels varie entre 300 et 500 pixels ce qui donne environ 2000 superpixels pour chaque image. Pour l'apparence, un algorithme des k-moyennes est utilisé sur les valeurs de couleur et de texture pour générer deux vocabulaires. L'un de 60 "mots" pour la texture et l'autre de 150 "mots" pour la couleur. La région d'intérêt est estimée en ne gardant que les échantillons 3D qui se projettent à l'intérieur des images pour toutes les vues. Environ 100k échantillons sont générés pour chaque jeu de données. Les paramètres λ_1, λ_2 et λ_3 sont fixés respectivement à 2.0, 4.0 et 0.05 dans tous les tests. Les H_i^F sont initialisés avec les pixels de la reprojection de la région d'intérêt 3D dans chaque image. Les histogrammes d'arrière-plan H_i^B sont initialisés avec les régions d'images à l'extérieur de cette région d'intérêt. Le temps de calcul dépend du nombre de points de vue. Par exemple, avec 10 caméras, chaque itération de l'algorithme prend en moyenne 10s avec notre implémentation C++ et la convergence est atteinte en moins de 10 itérations. Les tests ont été effectués sur une machine équipée d'un processeur Intel Core i7 à 2.3GHz avec 4GO de mémoire.

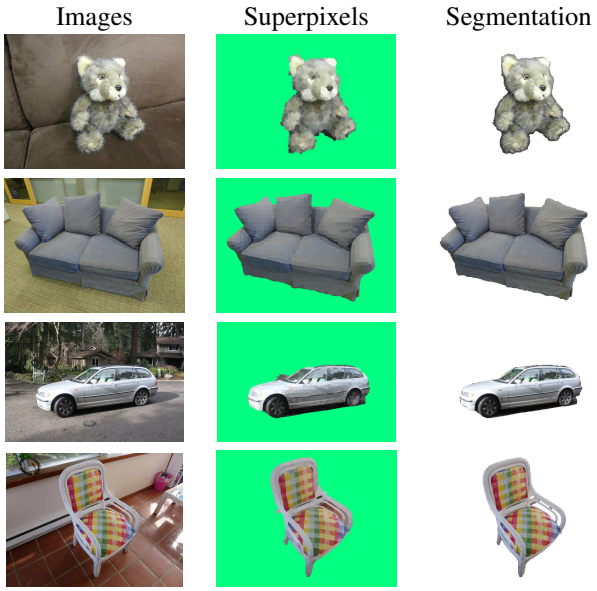
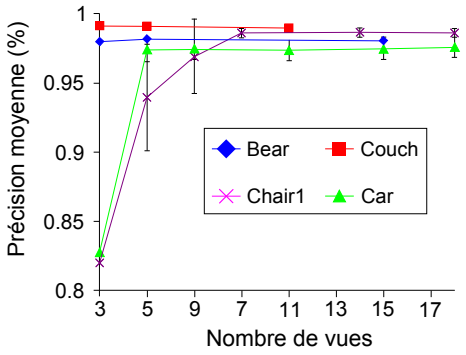


FIGURE 6 – Résultats sur BEAR (3 caméras), COUCH (3 caméras), CAR (5 caméras) et CHAIR (9 caméras). La première colonne montre l'une des images d'entrée. La deuxième et troisième colonne montrent respectivement la segmentation au niveau superpixel et au niveau pixel.

6.2 Résultats qualitatifs

Pour valider l'approche, une douzaine de jeux de données ont été utilisés. Il est à noter que dans la littérature existante très peu de données sont disponibles en libre accès, rendant les tests comparatifs difficiles. Certaines images ont été obtenues auprès des auteurs de [13] : COUCH, BEAR, CAR, CHAIR1. Ces données sont utilisées pour une évaluation qualitative et quantitative. D'autres images ont aussi été utilisées ici, telles que : BUSTE de [14] et PLANT [1]. Les figures 6 à 9 montrent les résultats de notre méthode sur les différents jeux de données : segmentation au niveau superpixel et le résultat final au niveau pixel. On remarquera en particulier la robustesse de l'approche à la réduction du nombre de points de vue. Sur la séquence PLANT, 3 caméras fortement espacées suffisent pour obtenir une segmentation de très bonne qualité, que les méthodes de

1. http ://vision.in.tum.de/data/datasets/rgbd-dataset



Séquence	Notre Méthode		Kowdle [13]	Djelouah [6]	Vicente [22]
Couch	3 99.1 ± 0.2	11 99.0 ± 0.2	11 99.6 ± 0.1	11 98.8 ± 08	non disponible
Bear	3 98.0 ± 1.0	15 98.0 ± 1.0	15 98.8 ± 0.4	15 98.8 ± 0.4	non disponible
Car	5 97.4 ± 0.8	44 97.0 ± 0.8	44 98.0 ± 0.7	44 0*	44 91.4 ± 4.3
Chair1	9 98.6 ± 0.3	18 98.6 ± 0.3	18 99.2 ± 0.4	18 88.0 ± 2.0	18 86.9 ± 7.8

(*) L'objet d'intérêt n'est pas identifié pour ce jeu de données.

FIGURE 7 – Évaluation quantitative de notre approche sur une scène statique. Le graphe montre la qualité des résultats de notre approche en fonction du nombre de vues utilisées. La tableau de droite montre la comparaison avec les méthodes de l'état de l'art (*nombre de vues*, *Précision*). La méthode proposée atteint la même qualité de résultats mais nécessite l'utilisation de beaucoup moins de points de vue.

l'état de l'art nécessitant un grand nombre de points de vue (e.g. [13]) ne seraient pas en mesure de fournir.

Images 8 vues 13 vues 26 vues

FIGURE 8 – Résultats sur la séquence BUSTE avec différents nombres de points de vue. Avec 8 caméras, la table apparait dans toutes les vues. Avec 13 caméras, certaines vues éliminent la table de l'avant-plan. Enfin, avec toutes les vues, les éléments noirs de l'arrière-plan apparaissent proches du socle noir de la statue qui est alors lui aussi classé comme arrière-plan.

Image Notre méthode Djelouah [6]

FIGURE 9 – Résultats sur la séquence PLANT (3 vues) et comparaison qualitative avec [6]. Notre approche bénéficie d'un modèle d'apparence plus riche et des contraintes de cohérence intra-image.

Dans des scénarios complexes, comme dans la Fig. 9, les approches reposant uniquement sur la couleur [6] ne parviennent pas à segmenter l'objet d'intérêt, tandis que notre méthode bénéficie d'un modèle d'apparence plus riche. Fig. 8 montre que ce qui est considéré comme objet d'intérêt dépend des points vue utilisés. Au départ, avec 8 caméras, la table fait partie de l'avant-plan. Lorsque des points de vue supplémentaires sont ajoutés, la table n'est plus entièrement vue dans toutes les images et elle est donc classée comme fond. En utilisant toutes les vues, dans plusieurs caméras les éléments d'arrière-plan noirs sont très proches du socle de la statue. Le socle est alors lui aussi segmenté comme fond et seule la statue reste comme objet d'intérêt vu par toutes les caméras.

6.3 Évaluations et comparaisons

Pour l'évaluation quantitative nous reprenons le protocole utilisé dans [13] qui considère la précision comme mesure de qualité de la segmentation. La précision est définie comme la proportion de pixels correctement classés (Fig. 7). Nous évaluons ici la sensibilité de la méthode proposée à la variation du nombre de caméras utilisées. Nous comparons aussi la qualité des résultats obtenus avec les méthodes de l'état de l'art [6, 13, 22]. Pour chaque valeur n du nombre de caméras, la précision correspond à une valeur moyenne sur 10 sous-ensembles de n vues aléatoirement choisies.

Les résultats quantitatifs Fig. 7 montrent clairement la robustesse de l'approche à la variation du nombre de caméras utilisées. On notera en particulier les très bons résultats sur les séquences CAR et CHAIR1, malgré le très faible nombre de vues utilisées et les ambiguïtés sur les couleurs entre fond et forme.

La différence de précision avec [13] s'explique par l'utilisation de l'information de profondeur et d'une reconstruction par plan de la scène dans [13]. Toutefois, cette précision s'obtient au prix d'un très grande nombre de vues pour l'estimation des cartes de disparité par stéréo.

7 Conclusion

Nous avons présenté une nouvelle approche au problème de la segmentation multi-vues. Cette approche est basée sur une coupe de graphe itérative et conjointe sur l'ensemble des vues. L'ensemble des contraintes intra-vue et inter-vues est exprimé dans un seul modèle de type champ aléatoire de Markov. L'approche s'avère robuste à l'utilisation d'un petit nombre de points de vue très éloignés les uns des autres et montre de bons résultats sur des séquences complexes. Nous pensons que ce schéma de segmentation constitue une base solide pour l'extension à la segmentation de séquences vidéo multi-vues.

Références

[1] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Susstrunk. SLIC Superpixels Compared to State-of-the-art Superpixel Methods. *IEEE PAMI*, 2012.

[2] D. Batra, A. Kowdle, D. Parikh, J. Luo, and T. Chen. icoseg : Interactive co-segmentation with intelligent scribble guidance. In *CVPR*, 2010.

[3] Y. Boykov and V. Kolmogorov. An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. *IEEE PAMI*, 2004.

[4] N. Campbell, G. Vogiatzis, C. Hernandez, and R. Cipolla. Automatic object segmentation from calibrated images. In *Visual Media Production (CVMP)*, 2011.

[5] N. D. F. Campbell, G. Vogiatzis, C. Hernández, and R. Cipolla. Automatic 3d object segmentation in multiple views using volumetric graph-cuts. *Image Vision Comput.*, 2010.

[6] A. Djelouah, J.-S. Franco, E. Boyer, F. L. Clerc, and P. Pérez. N-tuple color segmentation for multi-view silhouette extraction. In *ECCV*, 2012.

[7] B. Goldlücke and M. A. Magnor. Joint 3d-reconstruction and background separation in multiple views using graph cuts. In *CVPR*, 2003.

[8] J.-Y. Guillemaut and A. Hilton. Joint multi-layer segmentation and reconstruction for free-viewpoint video applications. *IJCV*, 2011.

[9] N. Hasler, B. Rosenhahn, T. Thormählen, M. Wand, J. Gall, and H. P. Seidel. Markerless motion capture with unsynchronized moving cameras. In *CVPR*, 2009.

[10] D. S. Hochbaum and V. Singh. An efficient algorithm for co-segmentation. In *ICCV*, 2009.

[11] A. Joulin, F. R. Bach, and J. Ponce. Discriminative clustering for image co-segmentation. In *CVPR*, 2010.

[12] K. Kolev, T. Brox, and D. Cremers. Fast joint estimation of silhouettes and dense 3d geometry from multiple images. *IEEE PAMI*, 2011.

[13] A. Kowdle, S. N. Sinha, and R. Szeliski. Multiple view object cosegmentation using appearance and stereo cues. In *ECCV*, 2012.

[14] W. Lee, W. Woo, and E. Boyer. Silhouette Segmentation in Multiple Views. *IEEE PAMI*, 2010.

[15] S. Nobuhara, Y. Tsuda, I. Ohama, and T. Matsuyama. Multi-viewpoint silhouette extraction with 3d context-aware error detection, correction, and shadow suppression. *Information and Media Technologies*, 2009.

[16] C. Reinbacher, M. Rother, and H. Bischof. Fast variational multi-view segmentation through backprojection of spatial constraints. *Image and Vision Computing*, 2012.

[17] C. Rother, V. Kolmogorov, and A. Blake. "grabcut" : interactive foreground extraction using iterated graph cuts. In *ACM SIGGRAPH*, 2004.

[18] C. Rother, T. Minka, A. Blake, and V. Kolmogorov. Cosegmentation of image pairs by histogram matching - incorporating a global constraint into mrfs. In *CVPR*, 2006.

[19] M. Sarim, A. Hilton, J.-Y. Guillemaut, H. Kim, and T. Takai. Wide-baseline multi-view video segmentation for 3d reconstruction. In *3DVP*, 2010.

[20] N. Snavely, S. M. Seitz, and R. Szeliski. Modeling the world from internet photo collections. *Int. J. Comput. Vision*, 2008.

[21] M. Sormann, C. Zach, and K. Karner. Graph cut based multiple view segmentation for 3d reconstruction. In *3DPVT*, 2006.

[22] S. Vicente, C. Rother, and V. Kolmogorov. Object cosegmentation. In *CVPR*, 2011.

[23] J. Xiao, J. Wang, P. Tan, and L. Quan. Joint affinity propagation for multiple view segmentation. In *ICCV*, 2007.

[24] G. Zeng and L. Quan. Silhouette extraction from multiple images of an unknown background. In *ACCV*, 2004.